

University of Groningen

Big data, but are we ready?

Trelles, Oswaldo; Prins, Pjotr; Snir, Marc; Jansen, Ritsert C.

Published in:
Nature Reviews Genetics

DOI:
[10.1038/nrg2857-c1](https://doi.org/10.1038/nrg2857-c1)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2011

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Trelles, O., Prins, P., Snir, M., & Jansen, R. C. (2011). Big data, but are we ready? *Nature Reviews Genetics*, 12(3), 224-224. <https://doi.org/10.1038/nrg2857-c1>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Big data, but are we ready?

Oswaldo Trelles, Pjotr Prins, Marc Snir and Ritsert C. Jansen

We welcome the timely Review by Schadt *et al.* (Computational solutions to large-scale data management and analysis. *Nature Rev. Genet.* **11**, 647–657 (2010))¹, which presents cloud and heterogeneous computing as solutions for tackling large-scale and high-dimensional data sets. These technologies have been around for years, raising the question: why are they not used more often in bioinformatics? The answer is that, apart from introducing complexity, they quickly break down when a large amount of data is communicated between computing nodes.

In their Review, Schadt and colleagues state that computational analysis in biology is high-dimensional, and predict that petabytes, even exabytes, of data will be soon stored and analysed. We agree with this predicted scenario and illustrate, through a simple calculation, how suitable current computational technologies really are for such large volumes of data.

Currently, it takes minimally 9 hours for each of 1,000 cloud nodes to process 500 GB,

at a cost of US\$3,000 (500 GB to 500 TB of total data). The bottleneck in this process is the input/output (IO) hardware that links data storage to the calculation node (FIG. 1). All nodes are idle for long periods, waiting for data to arrive from storage; shipping the data on a hard disk to the data storage would not resolve this bottleneck. We estimate that 1,000 cloud nodes each processing 1 petabyte (1 petabyte to 1 exabyte of total data) would take 2 years, and cost \$6,000,000.

A less expensive option would be to use heterogeneous computing, in which graphics processing units (GPUs) are used to boost speed. A similar calculation shows, however, that GPUs are idle 98% of the time when processing 500 GB of data. GPU performance rapidly degrades when large volumes of data are communicated, even with state-of-the-art disk arrays. Furthermore, GPUs are vector processors that are suitable for a subset of computational problems only.

Which is the best way forward? Computer systems that provide fast access to petabytes of data will be essential. Because high-dimensional large data sets exacerbate IO issues, the future lies in developing highly parallelized IO using the shortest possible path between storage and central processing units (CPUs). Examples of this trend are Oracle Exadata² and IBM Netezza³, which offer parallelized exabyte analysis by providing CPUs on the storage itself. Another trend for improving speed is the integration of photonics and electronics^{4,5}.

To fully exploit the parallelization of computation, bioinformaticians will also have to adopt new programming languages, tools and practices, because writing correct software for concurrent processing that is efficient and scalable is difficult^{6,7}. The popular R programming language, for example, has only limited support for writing parallelized software (see, for example, REF. 8). However, other languages^{9,10} can make parallel programming easier by, for example, abstracting threads¹¹ and shared memory⁷.

So, not only do cloud and heterogeneous computing suffer from severe hardware bottlenecks, they also introduce (unwanted) software complexity. It is our opinion that large multi-CPU computers are the preferred choice for handling big data. Future machines will integrate CPUs, vector

processors and random access memory (RAM) with parallel high-speed interconnections to optimize raw processor performance. Our calculations show that for petabyte- to exabyte-sized high-dimensional data, bioinformatics will require unprecedented fast storage and IO to perform calculations within an acceptable time frame.

Oswaldo Trelles is at the Computer Architecture Department, University of Malaga, Campus de Teatinos, E-29071, Spain.

Pjotr Prins and Ritsert C. Jansen are at the Groningen Bioinformatics Centre, University of Groningen, Nijenborgh 7, 9747 AG Groningen, The Netherlands.

Pjotr Prins is also at the Department of Nematology, Wageningen University, Droevendaalsesteeg 1, 6708 PB Wageningen, The Netherlands.

Marc Snir is at the Thomas M. Siebel Center for Computer Science, University of Illinois, MC258, 201 N. Goodwin Avenue, Urbana, Illinois 61801-2302, USA.

O.T. and P.P. contributed equally to this work.

Correspondence to R.C.J.
e-mail: r.c.jansen@rug.nl

Published online 8 February 2011

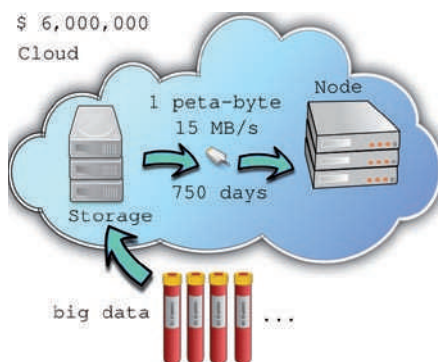


Figure 1 | Input/output bottleneck between data storage and calculation node. In our calculations, 1,000 computational nodes each processing 500 GB would take 9 hours (at a rate of 15 MB/s) using large nodes at US\$0.34/h. The total cost for a single analysis run would be $1,000 \times 9 \times 0.34 = \$3,060$. In reality, throughput will be lower because of competition for access to data storage caused by parallel processing. There are significant throughput instability and abnormal delay variations, even when the network is lightly utilized¹². In the illustrated example, 1,000 cloud nodes each processing a petabyte would take 750 days (at 15 MB/s) and cost $1,000 \times 750 \times 24 \times 0.34 = \$6,120,000$. Figure created using the Open Clip Art Library (<http://www.openclipart.org>).

- Schadt, E. E., Linderman, M. D., Sorenson, J., Lee, L. & Nolan, G. P. Computational solutions to large-scale data management and analysis. *Nature Rev. Genet.* **11**, 647–657 (2010).
- Grancher, E. Oracle and storage IOs, explanations and experience at CERN. *J. Phys. Conf. Ser.* **219**, 1–10 (2010).
- Davidson, G. S., Boyack, K. W., Zacharski, R. A., Helmreich, S. C. & Cowie, J. R. *Sandia Report SAND2006-3640: Data-centric computing with the Netezza architecture*. (Sandia National Laboratories, 2006).
- Vlasov, Y., Green, W. M. J. & Xia, F. High-throughput silicon nanophotonic wavelength-insensitive switch for on-chip optical networks. *Nature Photon.* **2**, 242–246 (2008).
- Reed, G. T. *Silicon Photonics: The State of the Art* (Wiley-Interscience, 2008).
- Mattson, T., Sanders, B. & Massingill, B. *Patterns for Parallel Programming* (Addison-Wesley Professional, 2004).
- Harris, T. *et al.* Transactional memory: an overview. *IEEE Micro* **27**, 8–29 (2007).
- Tierney, L., Rossini, A. J. & Li, N. Snow: a parallel computing framework for the R system. *Int. J. Parallel Prog.* **37**, 78–90 (2008).
- Kraus, J. M. & Kestler, H. A. Multi-core parallelization in Clojure: a case study. *Proc. 6th European Lisp Workshop* 8–17 (2009).
- Armstrong, J. *Programming Erlang: Software for a Concurrent World* (Pragmatic Bookshelf, 2007).
- Haller, P. & Odersky, M. Scala actors: unifying thread-based and event-based programming. *Theor. Comp. Sci.* **410**, 202–220 (2009).
- Wang, G. & Ng, T. E. S. The impact of virtualization on network performance of Amazon EC2 data center. *Proc. IEEE Infocom* 6 May 2010 (doi:10.1109/INFCOM.2010.5461931).

Acknowledgements

The authors are grateful for support by European Union grants FP7 PANACEA 222936, FP7 EURATRANS 241504 and cost action SYSGENET BM0901.

Competing interests statement

The authors declare no competing financial interests.

FURTHER INFORMATION

Ritsert C. Jansen's homepage: <http://www.rug.nl/gbic>

ALL LINKS ARE ACTIVE IN THE ONLINE PDF